AD-A070 476 MARYLAND UNIV COLLEGE PARK DEPT OF PHYSICS AND ASTRONOMY F/G 20/9
PLASMA MULTISTREAMING AND THE VIABILITY OF FLUID CODES, (U)
1976 W M MANHEIMER, M LAMPE, R CLARK N00014-75-C-0309
UNCLASSIFIED PUB-76-230 NL

1 OF 1
AD
A070476

END
DATE
FILMED
7 — 79
DDC

Code 6702

Preprint #604P001

PLASMA MULTISTREAMING AND THE VIABILITY OF FLUID CODES

by

W. M. Manheimer, M. Lampe, R. Clark and K. R. Chu
Naval Research Laboratory
Washington, D. C. 20375

and

P. C. Liewer
Department of Physics and Astronomy
University of Maryland
College Park, Maryland 20742

Physics Publication Number 76-230

1976

Work on this report was supported by ONR Contract N00014-75-C-0309 and/or N00014-67-A-0239 monitored by NRL 6702. 02.

UNIVERSITY OF MARYLAND
DEPARTMENT OF PHYSICS AND ASTRONOMY
COLLEGE PARK, MARYLAND

DDC PROCESSING DATA

PHOTOGRAPH

THIS SHEET

RETURN TO DDA-2 FOR FILE

II

LEVEL

1

INVENTORY

ADA070476

DDC ACCESSION NUMBER

Preprint 604P001, Physics Pub. 76-230

DOCUMENT IDENTIFICATION

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

DISTRIBUTION STATEMENT

Accession For	
NTIS GRA&I	X
DDC TAB	
Unannounced	
Justification	
By	
Distribution/	
Availability Codes	
Dist.	Avail and/or special
A	

DISTRIBUTION STAMP

DDC
RECEIVED
JUN 28 1979
D

DATE ACCESSIONED

79 06 27 309

DATE RECEIVED IN DDC

PHOTOGRAPH THIS SHEET

PLASMA MULTISTREAMING AND THE VIABILITY OF FLUID CODES

by

W. M. Manheimer, M. Lampe, R. Clark and K. R. Chu
Naval Research Laboratory
Washington, D. C. 20375

and

P. C. Liewer
Department of Physics and Astronomy
University of Maryland
College Park, Maryland 20742

and

Naval Research Laboratory
Washington, D. C. 20375

Work on this report was supported
by ONR Contract N00014-75-C-0309
and/or N00014-67-A-0239
monitored by NRL 6702.
02.

ABSTRACT

For plasmas in the collisionless regime, the absence of bulk viscosity permits multistreaming; if this occurs, fluid codes, which by definition prohibit multistreaming, give unphysical results. We show that a fluid code, using an energy propagation equation, will conserve energy but nevertheless can give totally incorrect results if multistreaming occurs. If a temperature propagation equation is used instead, the system continuously loses energy when multistreaming occurs. This energy nonconservation, which is completely independent of cell size and time step, can be used as an internal test for multistreaming, requiring no data outside the fluid code itself.

I. INTRODUCTION

Fluid codes, wherein each particle species is fully described by a single density, velocity and temperature at each position, have been widely used to calculate the flow of nearly collisionless plasmas. A few examples are electron beam - plasma interactions,[1] theta pinch implosions,[2,3] and the underdense regime of laser fusion pellets.[4,5] But a collisionless fluid has no bulk viscosity, and therefore it is possible for a single fluid species to counterstream against itself; indeed there is a natural tendency for counterstreaming to arise, particularly in high Mach number flows, even if the initial conditions are quite smooth and regular. When counterstreaming occurs, the subsequent fluid code solution becomes unphysical. Obviously the code will conceal the counterstreaming of the plasma, since each species is characterized by a single velocity at each location. Furthermore, the code will, in general, incorrectly predict even the evolution of average quantities, e.g., the mean flow velocity of the various streams.

It might be argued that counterstreaming is prevented by plasma instabilities. This is true in certain limited cases, but certainly not in general. For example, the electron-electron two stream instability[6] will thermalize the counterstreaming of electrons virtually instantaneously (on a fluid time scale), if the streaming velocity exceeds the thermal velocity and the densities of the two streams are comparable. Ion-ion instabilities[7,8] and modified two stream instabilities[9,10] act similarly (but more slowly) on ion counterstreaming, but only over a very limited range of parameters. For example, there appears to be no instability

that can prevent ion counterstreaming along the magnetic field, if the ion counterstreaming energy well exceeds the electron thermal energy.[11] Thus, although plasma instabilities may provide anomalous resistivity,[12-14] thermal conductivity,[15] and shear viscosity,[16] there is no quantitative theory indicating whether, when and how instabilities can provide anomalous bulk viscosity; there are surely many cases when no such effect exists.

In some cases where it is easy to distinguish between streams, e.g., plasma slugs flowing toward each other, or initial conditions involving multistreaming plasmas, the problem of counterstreaming has been handled by treating each stream as a distinct fluid species.[17] In general, however, one cannot predict when and where, in a fluid simulation, counterstreaming will occur within a given species. Multistreaming of a collisionless fluid arises from the natural tendency of velocity gradients to steepen as the flow evolves.[18] In traditional fluid codes,[19,20] this steepening causes numerical instabilities when the velocity gradient gets too large. Two techniques can be used to suppress these instabilities and induce well-defined, nonmultistreaming evolution of the numerical model: either introduction of an artificial (Von Neumann) viscosity of sufficient magnitude, or use of flux corrected transport[21] (FCT), a recent improvement in fluid codes. However, either technique terminates steepening of the flow by forming a strong shock (across only a few grid cells in the FCT case) rather than by multistreaming. The subsequent evolution can be quite unphysical. Thus, if we can argue that a fluid is subject to a small but nonvanishing bulk viscosity, either classical or anomalous, which prevents multistreaming, then proper use of FCT will provide a treatment that is physically reasonable. If, on the other hand, we have

decided that the bulk viscosity is absent, there is a clear need for a test to decide whether multistreaming occurs; if it does, the fluid code is simply not a viable tool. The main purpose of this paper is to give a simple internal test, applicable to nearly any fluid code and requiring no data except for the code runs themselves, to determine whether counterstreaming of bulk-viscosity-free fluids has occurred.

The structure of the paper is as follows. In Sec. II we use a simple analytic model of a cold fluid to review the origin of multistreaming and the difficulties it presents. In Sec. III we show that the energy which should go into multistreaming in a numerical code is lost from the system, in a way that is completely independent of the grid size and time step, if energy is propagated in the code by means of a temperature transport equation. If an energy transport equation is used instead, the multistreaming energy is not lost, but rather is thermalized; nevertheless, the solution is, in general, erroneous if there is no bulk viscosity. We show that these phenomena can be used as a test to indicate that the actual physical system multistreams, and thus, that the fluid code model has broken down.

In Sec. IV we present three examples, drawn from subjects of current research interest, of the breakdown of fluid code runs due to the onset of multistreaming. The application of the energy conservation test demonstrates multistreaming in all cases. Analytic solutions, test particles, and parallel use of a hybrid Vlasov-fluid code are used to demonstrate counterstreaming in the three cases, and prove the validity of the energy conservation test for these examples. The examples are the interpenetration of two counterstreaming plasma slugs, steepening of a theta-pinch implosion, and expansion of a heated plasma column into a surrounding cool plasma.

A brief summary is presented in Sec. V.

II. THE PROBLEM OF MULTISTREAMING

To demonstrate the way in which multistreaming of a collisionless plasma can arise out of seemingly innocent conditions, we consider the simplified version of the Euler equation for the one-dimensional flow of a force-free, cold (i.e., pressure-free) fluid with velocity u,

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0 \tag{1}$$

It is well known that this equation produces a singularity $\partial u/\partial x = \infty$, after a time $t_s = \left(\mathrm{Max}\, \frac{\partial u(t=0)}{\partial x}\right)^{-1}$. For longer times, the solution becomes multiple-valued. To see this, note that any function of x-ut is a solution to Eq. (1); thus u(x,t) is determined by the functional equation

$$u(x,t) = u[x - u(x,t)t] \; . \tag{2}$$

For instance, assume that $u(x,t=0) = u_0 \cos kx$. Then $u(x,t>0)$ is found by solving

$$u(x,t) = u_0 \cos\{k[x-u(x,t)t]\} \; . \tag{3}$$

Equation (3) is amenable to graphical solution at any given values of x and t, as shown in Fig. 1 at x=0, and for a succession of different times: at the initial time t_0, at an early time t_1 when the solution is single-valued at x=0, at a later time t_2 when it first becomes singular at x=0, and at a still later time t_3, when it is multiple-valued. Figure 2 shows plots of u(x) at times t_0, t_1, t_2, t_3.

One might be tempted to argue that the solutions shown in Figs. 1 and 2 cannot be the real physical solutions, because they are multiple-valued. A little thought reveals that this is not so: if the particles

are genuinely non-interacting (and therefore there is no bulk viscosity), as assumed in Eq. (1), the faster fluid elements will overtake and stream through the slower ones. For example, a Vlasov or particle code solution of the cold noninteracting particle problem would clearly show the phase space distribution evolving exactly as in Fig. 2. In fact a series of recent simulation and theoretical papers[22-27] have shown the evolution of multistreaming in collisionless plasmas, and have shown that it can play a key role in shock formation, i.e., fluid flow can steepen to form a shock, even in the absence of any true dissipation mechanism; multistreaming plays the role of a dissipation mechanism in such a laminar collisionless shock.

Let us now consider the solution of Eq. (1) by means of a finite difference scheme code, as shown in Fig. 3. The original cosine disturbance evolves as in Fig. 2 until the derivative $\partial u/\partial x$ becomes singular at $x = \pi/2k \equiv x_0$. In finite difference form, Eq. (1) says that the change of u in a grid cell is the total velocity flux into that cell, multiplied by the time step. By symmetry, the net flux into the cell centered on x_0 is always zero, and thus $u(x_0) = 0$ at all times, i.e., positive u flows in from the left and negative u from the right. These two contributions add up to zero, so $u(x_0)$ does not change. Thus a discontinuity, i.e., a shock, appears at x_0, as the velocity profile steepens to a sawtooth (time t_2). Subsequently (time t_3), the slope of the continuous portions of the u(x) curve decreases, as momentum flows into the discontinuity at x_0 from both sides, i.e., the shock position is stationary, but the shock amplitude slowly decreases in time.

The essential difference between the analytic solution and the finite difference (fluid code-like) solution is that the forward

and backward flux continue onward across the singularity in the former, but they mix in the central grid cell in the latter. This yields the single-valued solution that is imposed by the code formulation. The code thus introduces something like an artificial viscosity which prohibits multistreaming across discontinuities. Clearly Eq. (1) is a simplified model, but we shall see that it exhibits the essential phenomena that arise in a complete set of fluid equations that propagate density, velocity and temperature. In particular, we note that the total energy in the system steadily decreases after the singularity is formed.

Clearly the actual analytic solution to Eq. (1) and its solution by a fluid code become completely different after the actual solution becomes multiple-valued. Figure 3 does not even give a correct representation of the mean velocity obtained by averaging the various streams in Fig. 2, as is clear from the fact that the discontinuities in Fig. 3 are always located at the fixed points $x = \pm\pi/2k$, $3\pi/2k$, etc., while the discontinuities in Fig. 2 (the leading edges of the various streams) move along in time. This difference cannot be resolved by using a finer spatial or temporal grid, but rather is inherent in the assumed single-valued representation. Thus in any attempt to apply fluid codes to collisionless, unmagnetized ions, where multistreaming is permitted by the physics, it is essential to devise some test to see whether the fluid does multistream. It is apparent that any solution by a fluid code in the multistreaming regime must be regarded with the utmost skepticism.

III. A GENERAL TEST FOR MULTISTREAMING

In this section we show how to test the viability of a fluid code using only the code itself without recourse to any analytic solution. We shall see that the existence of multistreaming is implied if, independent of grid size and time step, energy is lost from the system when one uses a temperature propagation formulation, rather than an energy propagation formulation. Comparing the solutions of Eq. (1) shown in Figs. 2 and 3, this seems to be a reasonable result; all of the multistreaming energy shown in Fig. 2 is lost in Fig. 3. We begin by considering the problem of energy conservation in a viscous fluid, where multistreaming is forbidden, and then consider what happens when viscosity is absent.

The main point is that the fluid equation which propagates energy may be written in several different ways. For instance if there is no interaction between an ion species and any other species and there are no external fields, the energy equation for this species may be written in one of two forms; as an energy transport equation

$$\frac{\partial}{\partial t} \varepsilon + \frac{\partial}{\partial x} P = 0 \tag{4}$$

where ε is the total ion energy density and P is the total energy flux for this ion species, including contributions of all bulk and shear viscosity terms to the energy flux; or as a temperature transport equation

$$\frac{\partial}{\partial t} \varepsilon_T + \frac{\partial}{\partial x} P_T = Q \tag{5}$$

where ε_T is the <u>thermal</u> energy density, P_T is the <u>thermal</u> energy flux and Q denotes all source terms, including viscous heating $\nu(\frac{\partial u}{\partial x})^2$, and also $nT \frac{\partial u}{\partial x}$ terms.

If periodic boundary conditions are imposed on the system, and the equation is in conservative form [as is Eq. (4)], then standard numerical algorithms should conserve ε whether or not viscosity is present. (Note that the numerical solution of Eq. (1), which is in conservative form, does conserve $\int_0^L udx$, where L is the periodicity length, even though the computed solution is not at all accurate.) Steep gradients across a single cell can cause severe numerical difficulties in conventional codes, but can be managed by using flux corrected transport (FCT)[21] techniques.

If we maintain the cell size as a constant, but let the viscosity approach zero, the gradients across a shocked region become steeper and steeper. However, the gradient is bounded by the finite grid size, so the viscous heating rate in Eq. (5), $Q_{visc} = \nu(\frac{\partial u}{\partial x})^2$ approaches zero. Therefore, when using a temperature equation formulation, energy is lost as the viscosity approaches zero. Let us note however, that if the viscosity is held fixed at any nonzero value, so that multistreaming is forbidden, one can always choose a grid spacing sufficiently small so that Eq. (5) will conserve energy.

It is clear then, that if the bulk viscosity is rigorously zero, the temperature equation formulation of the fluid code cannot conserve energy across a sharp flow discontinuity, no matter how small the grid spacing and time step. Since multistreaming of the analytic solution is characterized by $\frac{\partial u}{\partial x} = \infty$, the unavoidable nonconservation of energy when using a temperature equation formulation is a sign that the plasma is multistreaming. This non-conservation can be used as a test for multistreaming; it indicates that a fluid code treatment is no longer physical. Of course energy conservation is never perfect in a code, due to numerical errors, but it is not difficult in practice

to distinguish energy nonconservation due to multistreaming, which has the following properties: (1) Energy only decreases, whereas numerical fluctuations in a code are typically random. (2) The rate of energy loss is insensitive to changes in cell size or time step, unlike numerical fluctuations or code breakdowns due to numerical instabilities. (3) When energy nonconservation is indicated, one can use other computer diagnostics to check carefully for velocity discontinuities (which are sometimes small) that reveal the presence, location, and often the nature of the counterstreaming. (4) Use of an energy transport equation instead of a temperature transport equation, in an otherwise identical code, will eliminate the energy nonconservation. The two formulations should give identical results until the onset of multistreaming, but may diverge greatly thereafter.

Since temperature-propagating fluid codes fail to conserve energy in multistreaming situations, they are obviously unphysical there, and no one would propose using them.[28] But substituting an energy propagation equation is not, in general, a solution to the problem, even though energy conservation is then assured. Such a code is never quantitatively correct in the presence of multistreaming, and in many cases it is totally wrong and misleading. We shall see in the next section that even the simple case of two interpenetrating noninteracting plasma slugs cannot be correctly treated with energy propagating fluid equations. Another vivid example is a collisionless shock. If a piston pushes into an unmagnetized plasma, the energy-propagating fluid equations will predict that a standard hydrodynamic shock, obeying the usual Rankine-Hugoniot jump conditions, runs ahead of the piston. Thus there is dissipation in the shock front (associated with the effective bulk viscosity introduced by the code), leading to greater

than adiabatic heating downstream. However it has been shown[27] that in an unmagnetized collisionless plasma, a shock exists only in a limited range of piston Mach numbers, and if the shock does exist, downstream heating is strictly adiabatic. "Dissipation" occurs upstream, in the form of an ion component which is accelerated by electric fields in the shock and piston, and runs ahead of the shock, counterstreaming against the ambient plasma. The difference between the actual shock and the type of shock produced by the fluid code is illustrated in Fig. 4. The moral to be drawn from this example is that an energy-propagating fluid code conserves <u>total</u> energy, but it may well deposit the energy in the wrong place, resulting in quite unphysical solutions. Therefore, if one is using an energy-propagating fluid code, but suspects that counterstreaming may occur, it is wise to test the validity of the run by switching to a temperature transport equation and checking for energy conservation.

IV. EXAMPLES OF COUNTERSTREAMING AND THE ENERGY CONSERVATION TEST

A. Counterstreaming Plasma Slugs

If two slugs of collisionless plasma stream toward each other, and we assume no anomalous coupling mechanisms are operative (as is physically reasonable in many situations), it is obvious that the slugs will meet, counterstream through each other and separate again with no heating of the plasma. The density profiles for such a situation, at a series of times, are shown in Fig. 5 (solid curves).

We have run this situation on a standard one-dimensional two-fluid (electrons and ions) code with FCT. Two different versions of the code were used. In the first, the ion temperature T_i was propagated by the equation

$$\frac{\partial T_i}{\partial t} + \frac{\partial}{\partial x}(u_i T_i) = -\frac{1}{3} T_i \frac{\partial u}{\partial x} + \frac{2}{3}\frac{Q_i}{n_i} , \tag{6}$$

where u_i is the ion fluid velocity, n_i is the ion density, and Q_i is the ion heating rate due to anomalous resistive heating (zero in this case, but significant in examples B and C). The ion energy density is then calculated as

$$\varepsilon_i = \frac{3}{2} n_i T_i + \frac{1}{2} n_i m_i u^2 . \tag{7}$$

In the second version of the code, the ion energy density was propagated directly, with the equation

$$\frac{\partial \varepsilon_i}{\partial t} + \frac{\partial}{\partial x} u_i (\varepsilon_i + n_i T_i) = Q_i . \tag{8}$$

In accordance with expectations, a singularity in $\partial u_i/\partial x$ arises when two slugs make contact, and persists thereafter. In the temperature propagation formulation, the total system energy begins to drop at

that time, and continues to fall linearly with time until all of the counterstreaming energy has vanished from the system (Fig. 6). Matter flows into and piles up in the grid cell containing the singularity, as shown in Fig. 5 (dashed curves). The results are obviously unphysical. (There is some compressional heating here, and at later times, the plasma does expand slowly).

When the same problem is run in the energy-propagating code formulation, energy is conserved, as expected, to within one part in 10^4 (Fig. 6). The sequence of density profiles for this case are plotted as the dot-dashed curves in Fig. 5. We note that the streaming energy that was lost in the temperature case is here converted into thermal energy, as if by a bulk viscosity. The resulting pressure prevents as much density buildup at the origin as occurred in the case of the temperature-propagating code. The hot plasma then begins to expand again, in a manner quite unlike the physical solution.

We note that the situation discussed in this example is applicable to a crucial stage of a fast theta pinch implosion. When imploding fluid elements from opposite ends of the device (θ different by π) reach the center, the fluid density and velocity along a line through the center look approximately as shown in Fig. 5 (t=0). Thus all of the counterstreaming ion energy will be lost if a temperature equation is used. For strong pistons, the ion temperature computed with fluid codes using a temperature equation is much less than that measured experimentally.[3] In Ref. 3 the code was never checked for conservation of total energy since there are many external sources and sinks. However, it does seem reasonable that the source of the problem is the tendency of the theta pinch to multistream. Thus the resolution may not be the use of simple fixes, or even necessarily in the use of an energy equation instead of a temperature

equation. Rather the problem may be in the nature of the fluid model itself, and its resolution must come from a hybrid fluid-particle code similar to that used in example C of this section.

B. Theta Pinch Implosion

We discuss here a much more subtle case in which counterstreaming grows directly out of the steepening of a collisionless theta-pinch implosion, rather than originating in the collision of oppositely flowing plasma elements. The calculations were run with a one-dimensional temperature propagating two-fluid code, i.e., Eq. (6), and the energy density $\varepsilon_i(x,t)$ is then given by Eq. (7). In addition, a parallel calculation was done using the energy-propagating Eq. (8) to calculate ε_i, but using n_i, $\underline{v}_i$, B, etc. as calculated via Eq. (6). The total ion energy calculated in the former way is designated E_T; the ion energy calculated in the latter way is designated E_E. To check for ion multistreaming, we can compare E_E to E_T: we expect $E_E = E_T$ prior to counterstreaming, but $E_E > E_T$ when counterstreaming has occurred. To prove that $E_E > E_T$ does indeed occur when counterstreaming occurs, a set of test particle ions moving in the electric field generated by the temperature-propagating fluid code was used as an additional diagnostic.

Results from one calculation are shown in Fig. 7: ion fluid density and magnetic field profiles as functions of radius x in Fig. 7a, and plots of test ion phase space density f(x,u) at the same time in Fig. 7b. At this time $E_T = 0.99\ E_E$. Since E_T and E_E agree well, the energy conservation test would predict no multistreaming. And in fact the test ion phase space plot (Fig. 7b) bears this out,

showing no multistreaming at any value of x.

Results from a slightly different calculation (density, applied magnetic field, and temperature were the same, but bias field was in the opposite direction to the applied field) are shown in Fig. 8. For this calculation at the time shown, $E_T = 0.965\ E_E$. Since the code normally conserves ion energy to within 1% (as in the run of Fig. 7), the energy test would indicate that the fluid wants to multistream. And indeed the test ion phase space at this time (Fig. 8b) verifies that this is the case.

In Fig. 9, results are presented at a later time, from the same calculation as Fig. 8. At this time, $E_T = 0.8\ E_E$, so 20% of the ion energy has been lost. Here the test ion phase space shows a much larger number of counterstreaming ions, and the fluid profile shows enormous steepening.

To further verify that the energy loss is caused by multistreaming, and not by effects associated with the grid size (e.g., gradients too large to handle within a given grid size), the case shown in Figs. 8,9 was rerun with a grid size twice as large. As might be expected, the discrepancy between E_E and E_T increased somewhat at early times, when no counterstreaming occurs, to 2.5% as compared to 1% for the finer grid. But at the time of Fig. 9, the coarser grid calculation showed

$$E_T \simeq .8\ E_E$$

as did the finer grid. Since halving the grid size more than doubled energy nonconservation in the absence of multistreaming, but had no noticeable effect on the much larger energy loss after counterstreaming occurred, this large loss cannot be attributed to effects of finite grid size.

C. Expansion of a Hot Plasma Column

In many situations, one species of particles will tend to multistream, while other species will not. For example, in the theta pinch implosion considered above, ions counterstream while electrons are expected to act as a well-behaved fluid. One way to deal with such a situation is to use a hybrid code with a Vlasov or particle treatment of the ions and a fluid treatment of the electrons.[24,29,30] Such a code is of course slower running than a pure fluid code, but typically much faster than a purely Vlasov or particle code.

We illustrate this point with a numerical study of the radial expansion of a cylinder of plasma with hot electrons and cold ions into a surrounding cold plasma of equal density, which has been reported in more detail elsewhere.[1] A uniform axial magnetic field B_z is present, with $\beta \equiv 8\pi n T_e/B_z^2 > 1$ in the hot central plasma but $\beta \ll 1$ in the cold outer plasma, and the outer plasma is bounded by a perfectly conducting metallic shell. Because of the pressure difference, the hot plasma expands into the cold plasma, converting part of the electron thermal energy into ion streaming energy. The plasma subsequently bounces inward after the magnetic flux is compressed against the metallic shell, and this cycle is repeated several times.

Previous work[27] has indicated that if a "piston" pushes into a collisionless plasma, the flow will multistream only if the piston is strong enough. In our case, the "piston strength" corresponds to β, the ratio of the hot plasma pressure to the magnetic pressure (which confines the cold plasma). Therefore, we have done a low-β run A(β=1.3) and a high-β run C(β=13) with a fluid code (described elsewhere[2,3]), using a temperature propagating equation in each case. We then did

two similar runs (B, β=2 and D, β=20) on a hybrid code (described elsewhere[29]) with fluid electrons and Vlasov ions (henceforth called the Vlasov code). (Because of the numerical peculiarities of each code,[1] it was inconvenient to run exactly the same initial conditions on each code, but the low-β cases run on each code are qualitatively similar, as are the high-β cases.)

Figure 10 shows the total energy, electron thermal energy, magnetic energy, ion streaming energy and ion thermal energy, as functions of time, for each of the four runs. We note that total energy is conserved, in the fluid code, to within 0.1% in the low-β case (over 4000 time steps), but about 15% of the energy is lost in the high-β case (over 2000 time steps). Thus the energy conservation test leads us to believe that multistreaming should occur in the high-β case, but not in the low-β case. This is confirmed by the Vlasov code: the ion phase space densities plotted in Fig. 11 show no multistreaming in the low-β case, but no fewer than seven distinct streams at a late time in the β=20 case. We notice also that time dependence of the various energy components, shown in Fig. 10a,b, is qualitatively similar in the low-β fluid and Vlasov runs, and in both cases is smooth and physically reasonable. This is an indication of the viability of the fluid code in this case, where no counterstreaming occurs, even though the plasma is collisionless. On the other hand, the time development of the fluid and Vlasov high-β runs is strikingly different (Fig. 10c,d), with the curves for the fluid run being jagged and unphysical. We note particularly that the ion "thermal" energy (in the Vlasov code, this is defined in terms of the velocity spread at each point in space, but is really largely counterstreaming energy rather than true thermal energy) remains small in the fluid run, but reaches a large value in the Vlasov run, indicating that counterstreaming energy is lost from the system in the fluid run.

V. CONCLUSIONS

We have shown that the multistreaming is a natural development in collisionless plasma flow, which should be anticipated in the absence of evidence to the contrary. If multistreaming does occur, fluid codes do not, in general, correctly model the subsequent evolution of the plasma. In the case of a fluid code with a temperature propagation equation, this becomes quite obvious, since the energy associated with multistreaming vanishes from the system. For an FCT fluid code with an energy propagation equation, the onset of multistreaming is masked, since the code will conserve energy. Nonetheless, the code results can become completely unphysical.

When fluid codes are used to model collisionless plasmas which are free of bulk viscosity, it is thus imperative to check for multistreaming, which is signaled by a monotonic decrease in total system energy in the temperature propagation formulation, which is independent of changes in cell size or time step, but which goes away if an energy propagation equation is substituted. If the test indicates multistreaming, then results from either formulation of the fluid code must be regarded with extreme skepticism.

ACKNOWLEDGMENTS

We would like to thank Drs. Jay P. Boris, Timothy P. Coffey, and John H. Gardner.

This work was supported by Defense Nuclear Agency and the Energy Research and Development Administration.

REFERENCES

1. K. R. Chu, R. Clark, M. Lampe, P. Liewer and W. Manheimer, Phys. Rev. Lett 35, 94 (1975).
2. P. C. Liewer and N. A. Krall, Phys. Fluids 16, 1953 (1973).
3. E. Oktay, A. W. DeSilva, P. C. Liewer, Y. G. Chen, H. R. Griem, R. Hess and N. A. Krall, Plasma Physics and Controlled Nuclear Fusion, Vol. III, 365 (1975).
4. J. S. Clarke, H. N. Fisher and R. J. Mason, Phys. Rev. Lett. 30, 89 (1973).
5. N. K. Winsor and D. A. Tidman, Phys. Rev. Lett. 31, 1040 (1971).
6. J. Dawson, Nucl. Fusion Suppl. 3, 1033 (1962).
7. D. A. Tidman and N. A. Krall, Shock Waves in Collisionless Plasmas, Chapter 7.6, Wiley & Sons, New York, 1971.
8. K. Papadopoulos, R. C. Davidson, J. M. Dawson, I. Haber, D. A. Hammer, N. A. Krall and R. Shanny, Phys. Fluids 14, 849 (1971).
9. N. A. Krall and P. C. Liewer, Phys. Rev. A, 2094 (1971).
10. E. Ott, J. B. McBride, J. H. Orens and J. P. Boris, Phys. Rev. Lett. 28, 88 (1972).
11. D. W. Forslund and C. R. Shonk, Phys. Rev. Lett 25, 281 (1970).
12. M. Lampe, W. M. Manheimer, J. B. McBride, J. H. Orens, K. Papadopoulos, R. Shanny and R. Sudan, Phys. Fluids 15, 662 (1972).
13. W. Manheimer and J. P. Boris, Phys. Rev. Lett. 28, 659 (1972).
14. S. P. Gary, J. Plasma Phys. 4, 753 (1970); P. C. Liewer and N. A. Krall, Phys. Rev. Lett. 30, 1242 (1973).
15. D. W. Forslund, J. Geophys. Res. 75, 17 (1970).
16. W. Horton, Phys. Rev. Lett. 28, 1506 (1972).

17. Even in these cases, if some particles can be transferred from one stream to another (e.g., picked up by anomalous transport mechanisms), one must devise special algorithms to model the situation. This is done, e.g., by K. Papadopoulos, R. W. Clark and C. E. Wagner, Proceedings of the Third Conference on Solar Wind, Ed. C. T. Russell, UCLA, 343-350 (1974).

18. Landau and Lifshitz, Fluid Mechanics, 894, Addison-Wesley, Reading, Mass. (1959).

19 J. Von Neumann and R. D. Richtmyer, J. Applied Phys. 21, 232 (1952).

20. R. D. Richmyer and K. W. Morton, Difference Methods for Initial Value Problems, 12.13 Wiley & Sons, New York (1957).

21. J. P. Boris and D. L. Book, J. Comput. Phys. 11, 38 (1973).

22. K. Papadopoulos, C. E. Wagner and I. Haber, Phys. Rev. Lett. 27, 986 (1971).

23. D. W. Forslund and C. R. Shonk, Phys. Rev. Lett. 25, 1699 (1970).

24. D. W. Forslund and J. P. Friedberg, Phys. Rev. Lett. 27, 1189 (1971).

25. P. H. Sakanaka, Phys. Fluids 14, 611 (1971).

26. R. Mason, Phys. Fluids 15, 1082 (1972).

27. W. Manheimer and I. Haber, Phys. Fluids 17, 706 (1974).

28. There are cases where the system of interest has many sources and sinks, so that checks on total energy may be inconvenient and one might not be aware of the breakdown, even in a temperature-propagating code.

29. J. Denavit, J. Comp. Phys. 9, 75 (1972).

30. P. C. Liewer, Proceedings of Seventh Conf. on Numerical Simulation of Plasmas, New York City, (1975).

Figure Captions

Fig. 1 Graphical solution of the equation $u=u_0\cos(kut)$ at four successive times $t=0$, t_1, t_2, t_3. The solutions at a given time t are the intersections of the curve $f = u_0\cos(kut)$ with the straight line $f=u=kut/kt$. At t_2 there are two solutions, and at t_3 there are five.

Fig. 2 Solutions $u(x,t)$ of Eq. (3) at the initial time and three subsequent times t_1, t_2, t_3.

Fig. 3 Solutions $u(x,t)$ of Eq. (1) as obtained by a finite difference scheme, at the initial time and three subsequent times t_1, t_2, t_3.

Fig. 4 (a) Phase space in the shock frame for a laminar, collisionless electrostatic shock. (b) Phase space for a viscous shock, e.g., as produced by a fluid code in the collisionless regime.

Fig. 5 Density profiles at four different times for counterstreaming collisionless plasma slugs. Solid curves are the actual analytic results, dashed curves are the results from a temperature-propagating fluid code, and dot-dashed curves are the results of an energy-propagating fluid code.

Fig. 6 Time dependence of the ion energy, for counterstreaming plasma slugs. The dashed curve is from the temperature-propagating fluid code; the dot-dashed line (constant ε_i) is from the energy-propagating fluid code.

Fig. 7 (a) Profiles of ion density n and magnetic field B for a non-counterstreaming θ-pinch case. (b) Ion phase space snapshot at the same time.

Fig. 8 (a) Profiles of n and B, for a reversed-bias θ-pinch implosion. (b) Ion phase space at the same time, showing the onset of counterstreaming.

Fig. 9 (a) Profiles of n and B from the same run as Fig. 8, but at a later time. (b) Ion phase space at the same time, showing strong counterstreaming.

Fig. 10 Time development of total energy W, electron and ion thermal energies W_e and W_i, magnetic energy W_B [or $\Delta W_B \equiv W_B - W_B(0)$], and ion streaming energy k_i, for the fluid simulations A(β=1.3) and C(β=13), and the Vlasov simulations B(β=2) and D(β=20).

Fig. 11 Ion phase space snapshots from Vlasov-ion simulations B and D.

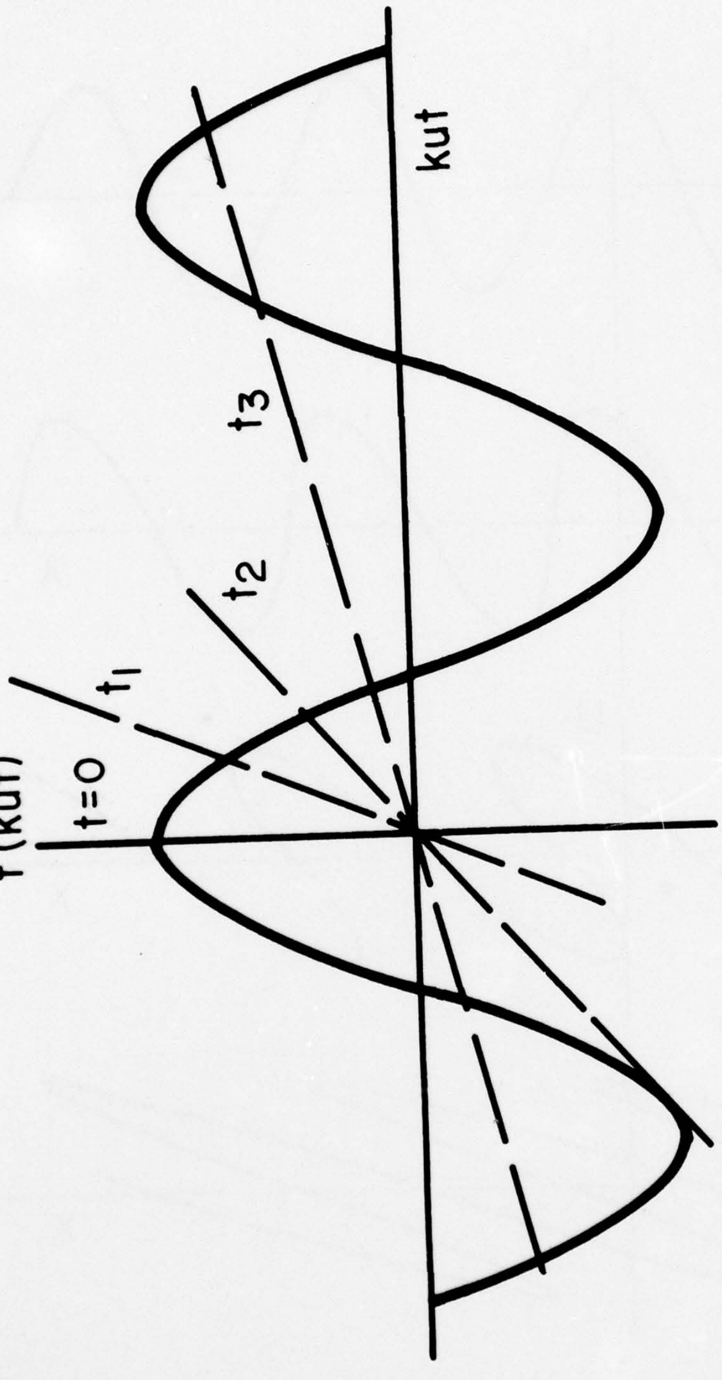


Fig. 1

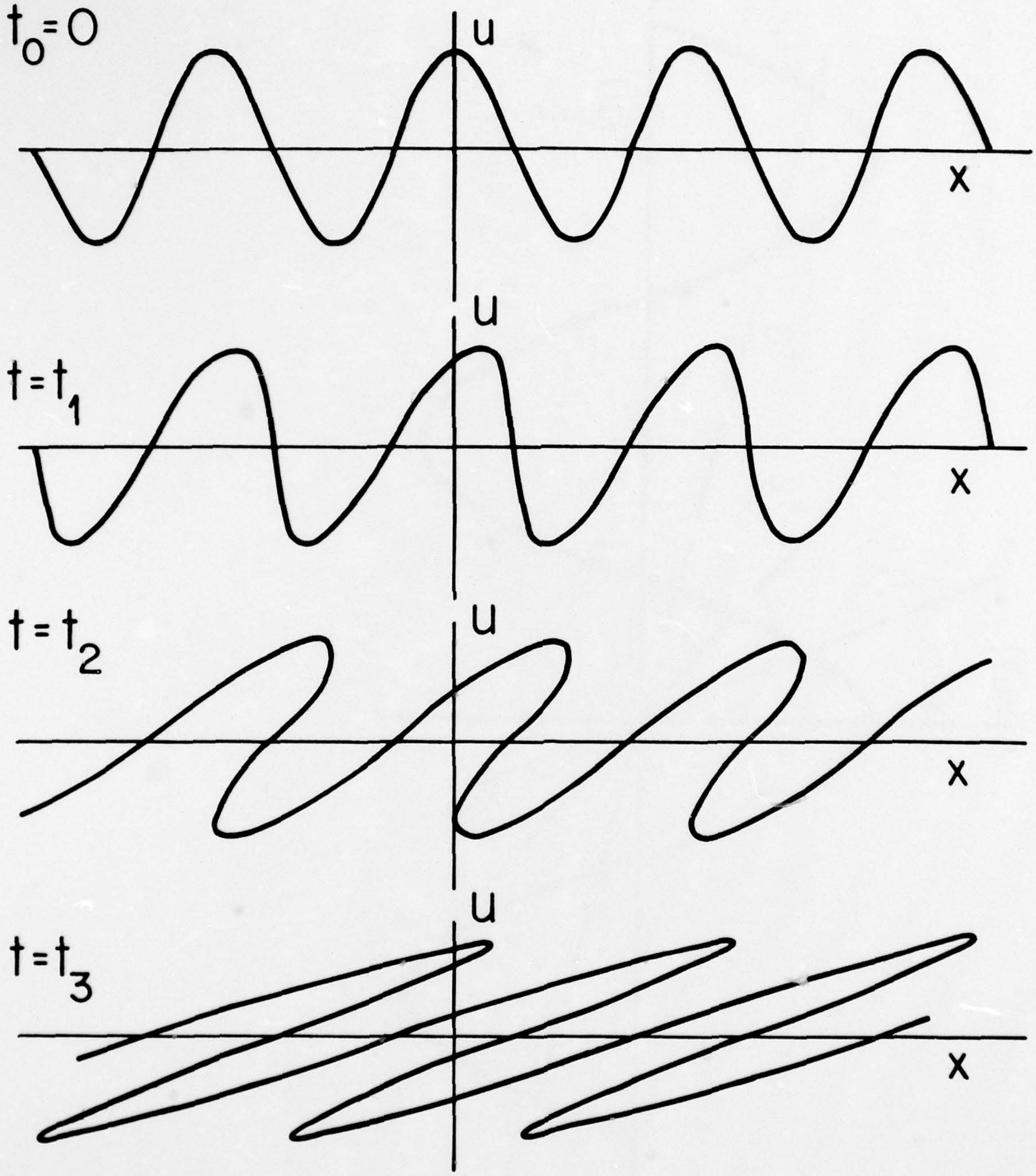


Fig. 2

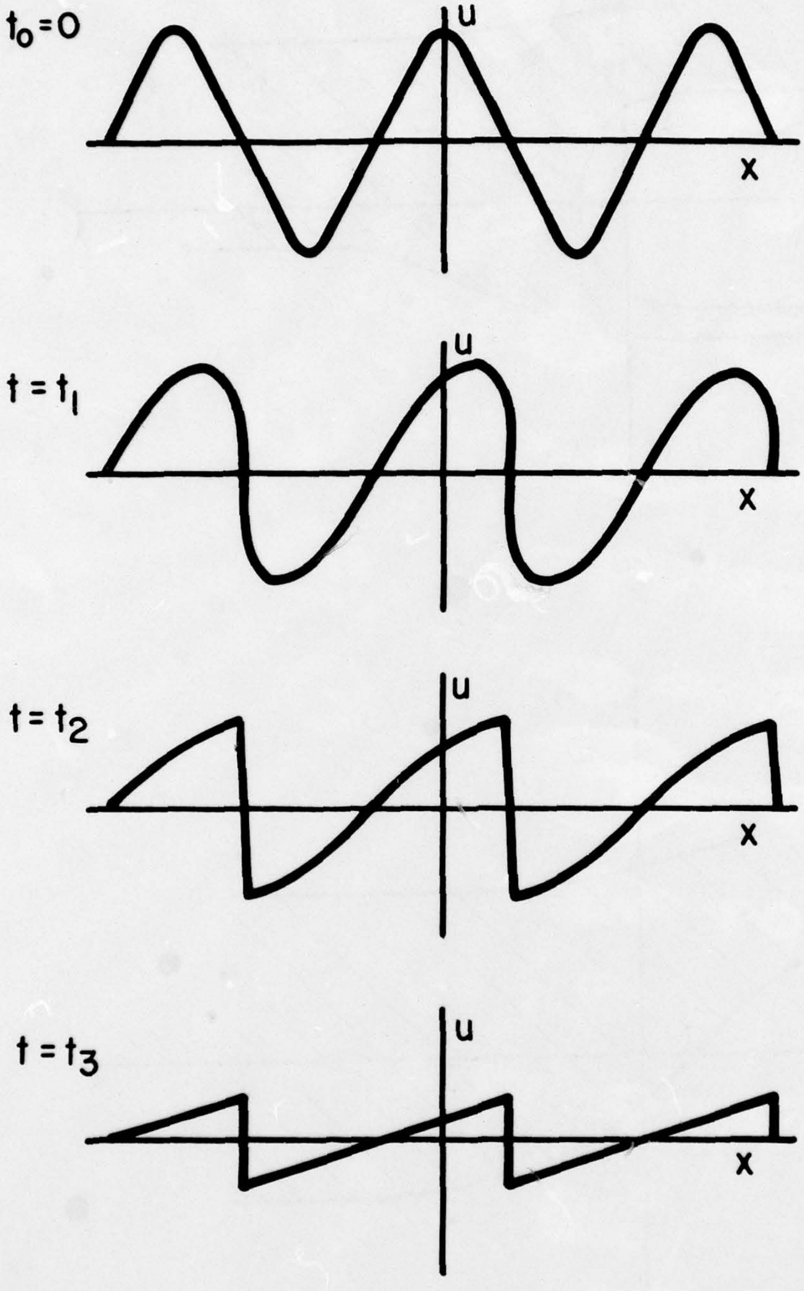


Fig. 3

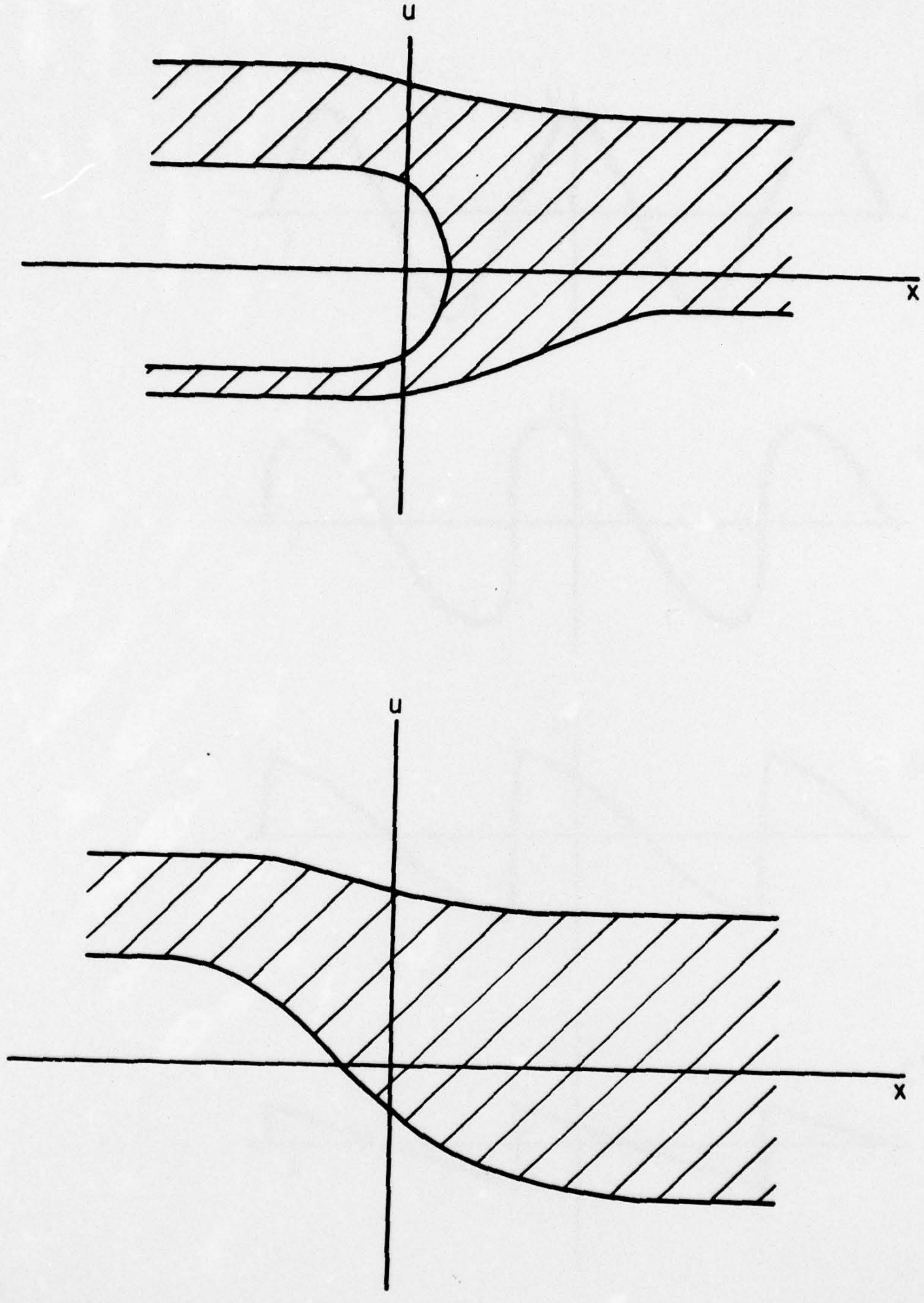


Fig. 4

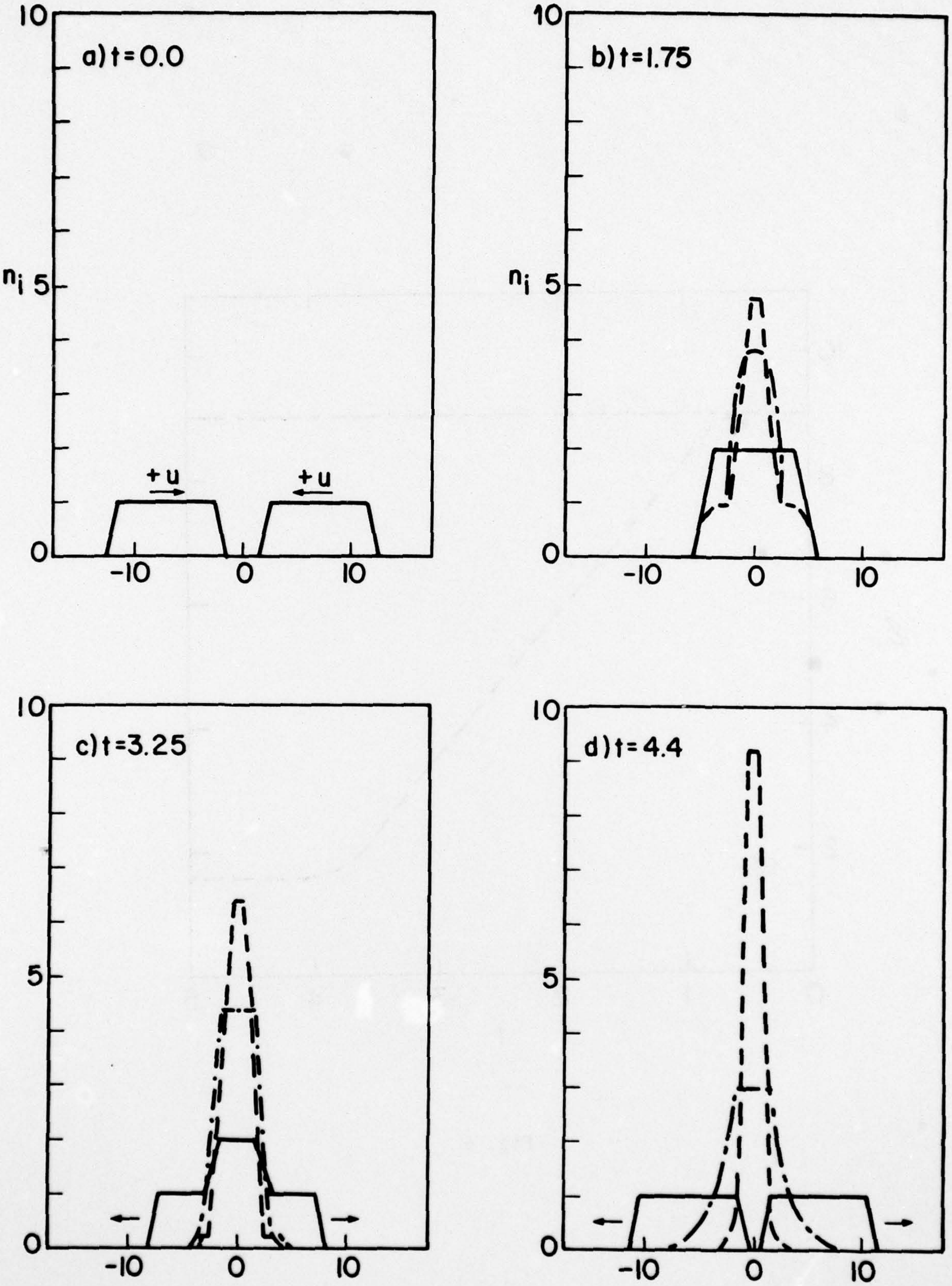


Fig. 5

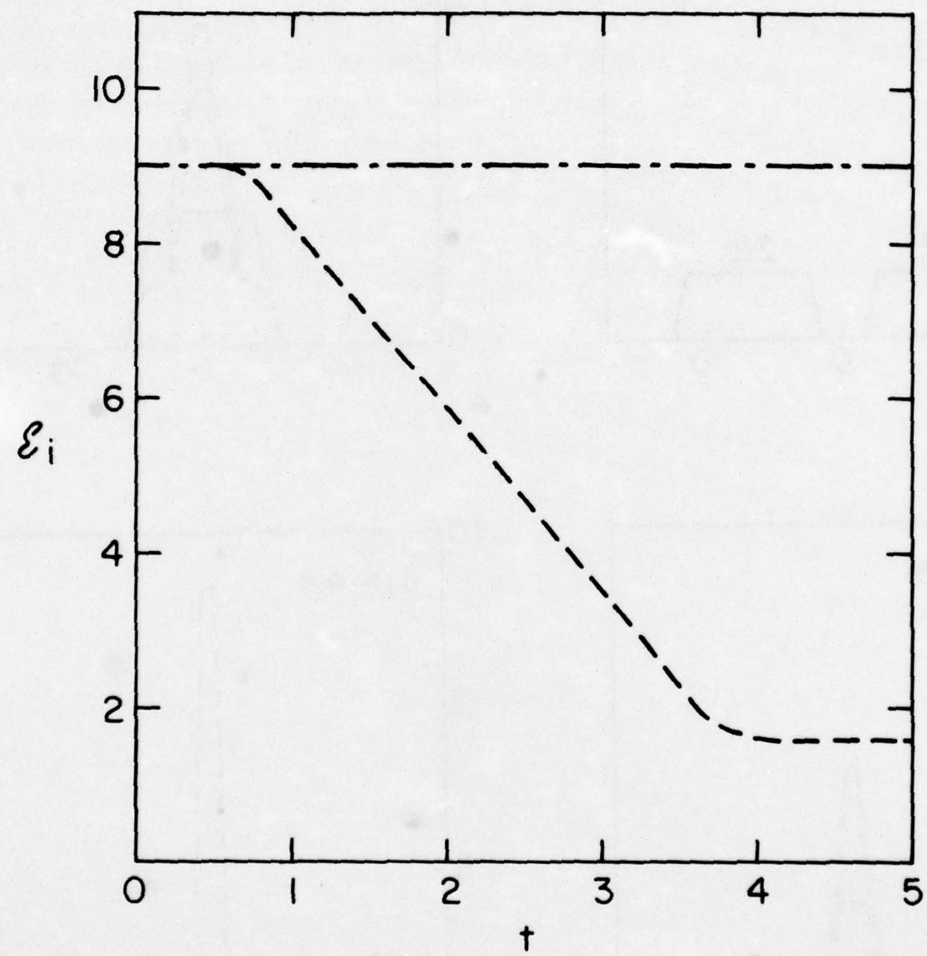
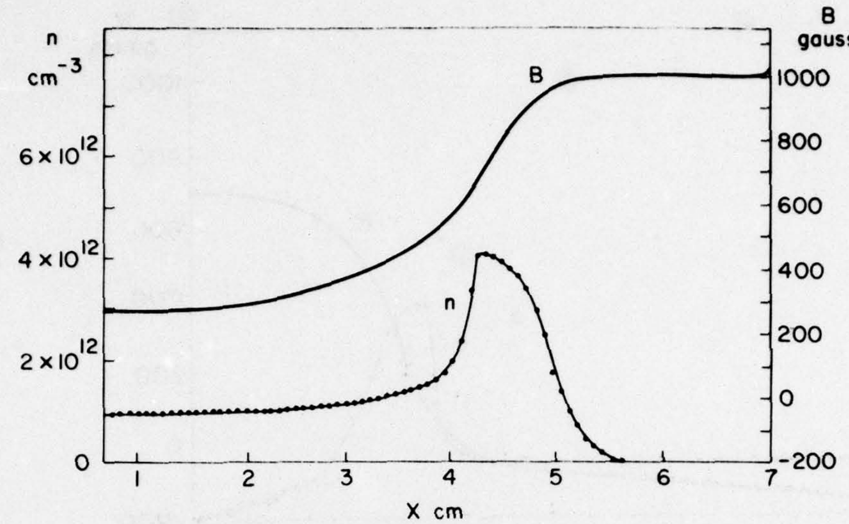


Fig. 6


n
cm^{-3}
B
gauss
B
n
6×10^{12}
4×10^{12}
2×10^{12}
0
1000
800
600
400
200
0
-200
1
2
3
4
5
6
7
X cm

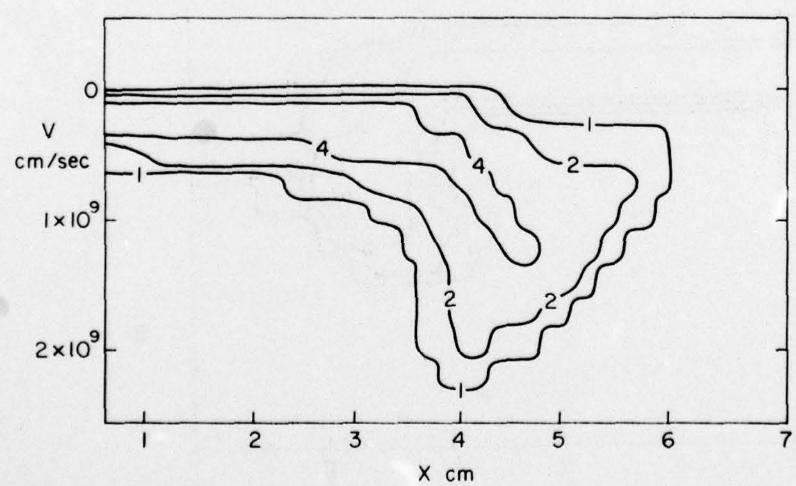

V
cm/sec
0
1×10^{9}
2×10^{9}
1
2
4
1
2
3
4
5
6
7
X cm

Fig. 7

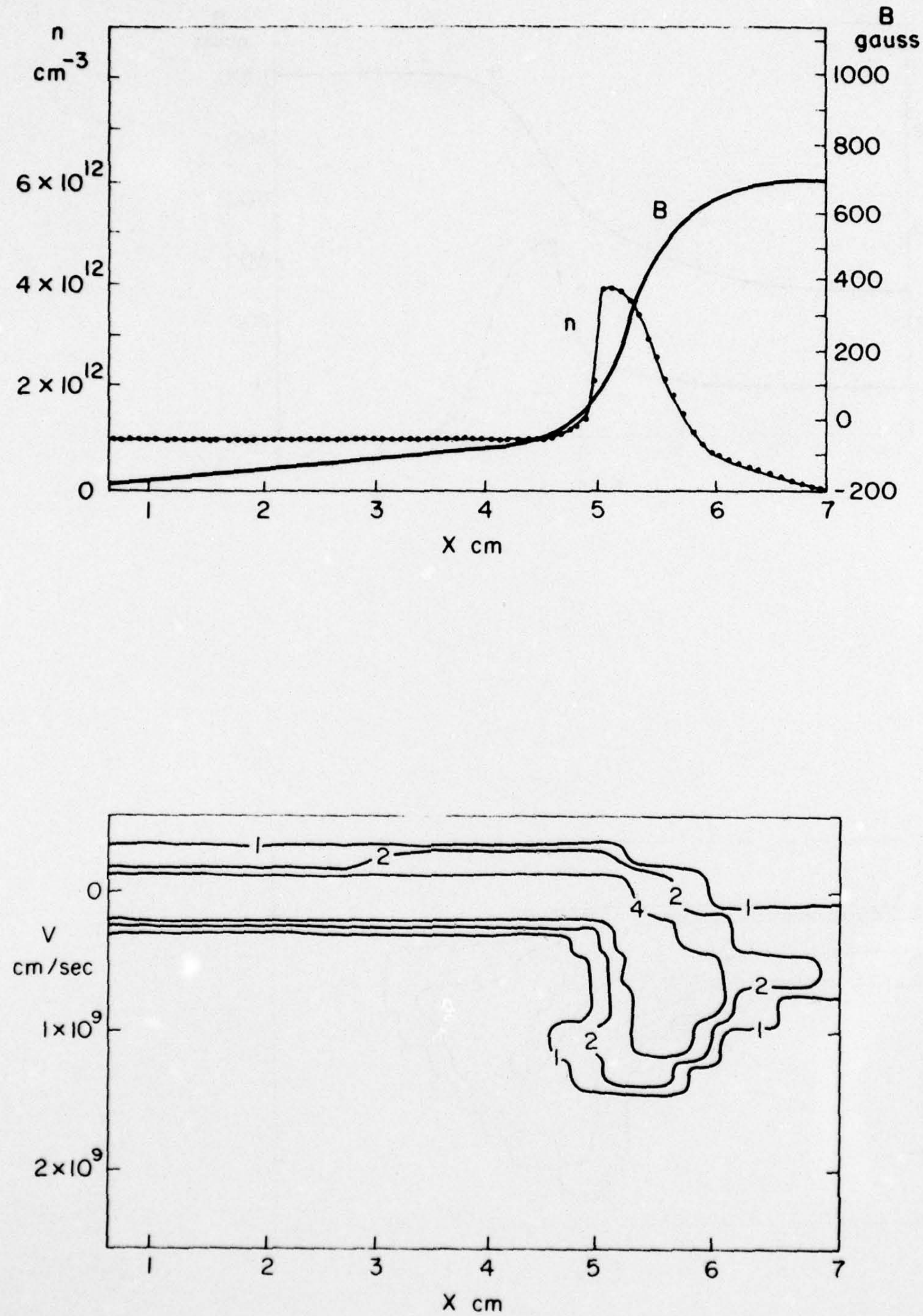
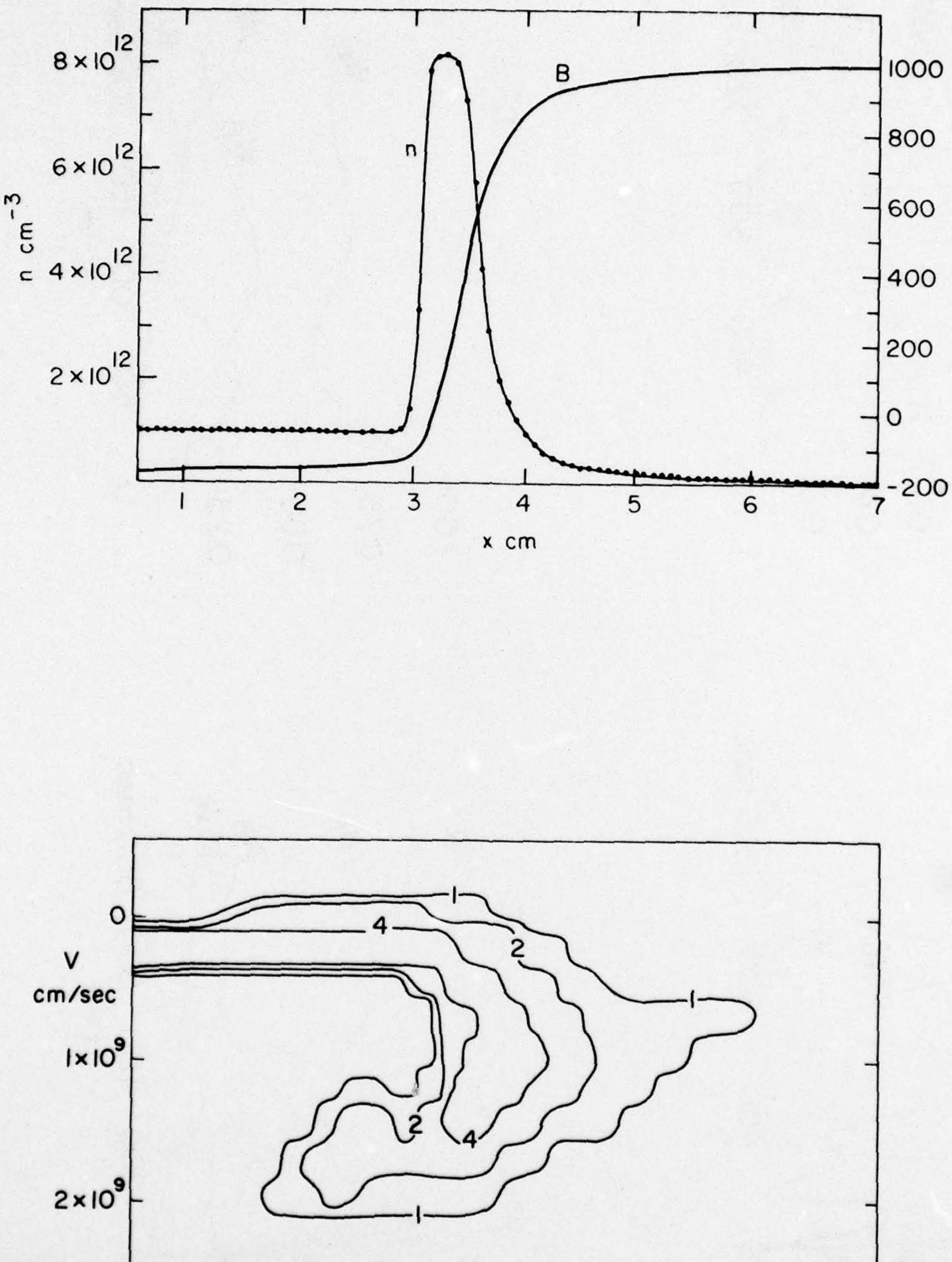
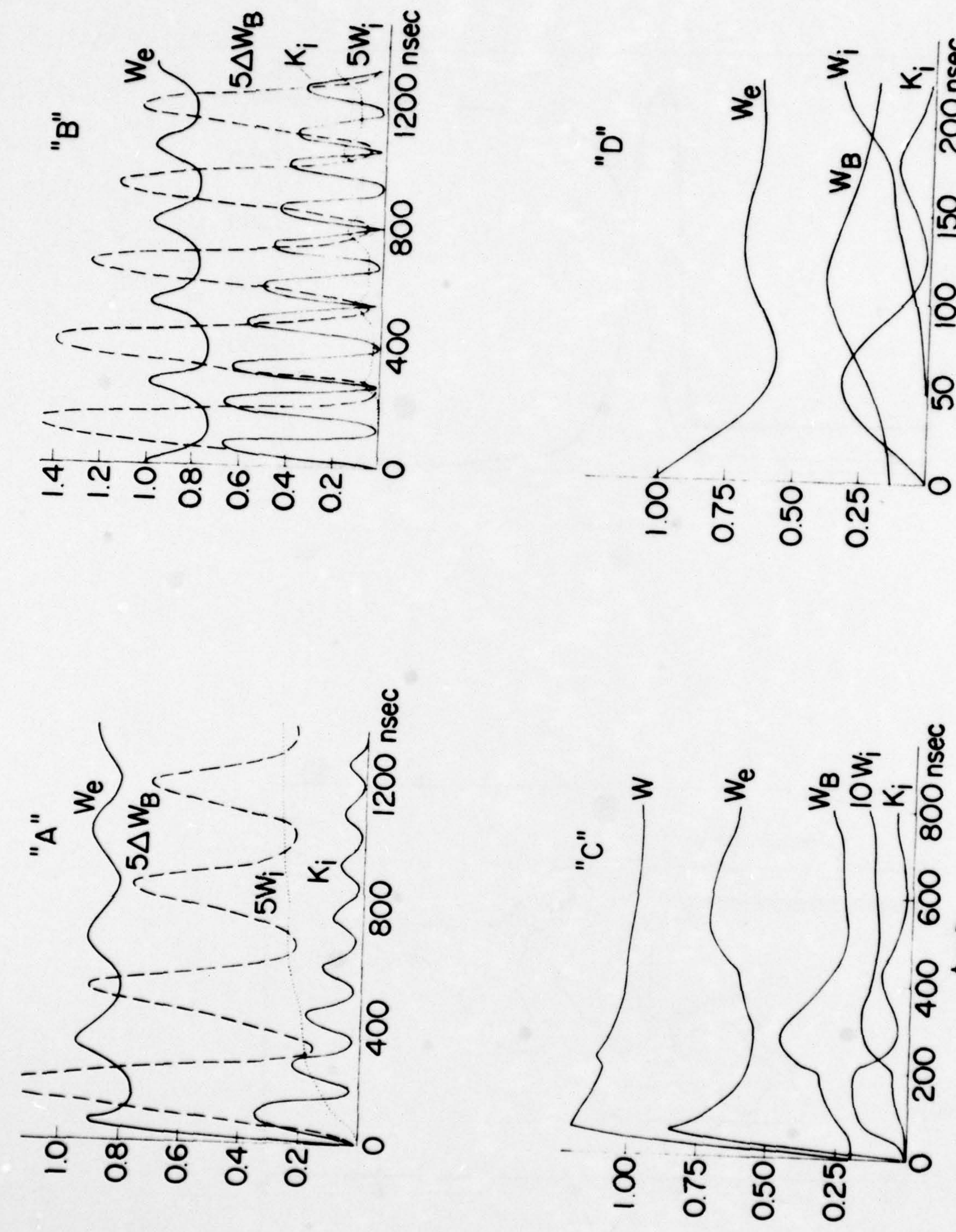

n
cm^{-3}
B
gauss
1000
800
600
400
200
0
-200
6×10^{12}
4×10^{12}
2×10^{12}
0
B
n
1
2
3
4
5
6
7
X cm
0
V
cm/sec
1×10^{9}
2×10^{9}
1
2
4
2
1
2
1
2
1
1
2
3
4
5
6
7
X cm

Fig. 8



8×10^12
6×10^12
4×10^12
2×10^12
n cm^-3
n
B
1000
800
600
400
200
0
-200
B gauss
1
2
3
4
5
6
7
x cm
0
V
cm/sec
1×10^9
2×10^9
1
2
4
1
2
4
1
1
2
3
4
5
6
7
X cm

Fig. 9



"A"
W_e
$5\Delta W_B$
$5W_i$
K_i
1200 nsec
"B"
W_e
$5\Delta W_B$
K_i
$5W_i$
1200 nsec
"C"
W
W_e
W_B
$10\,W_i$
K_i
800 nsec
t →
"D"
W_e
W_B
W_i
K_i
200 nsec
t →

Fig. 10

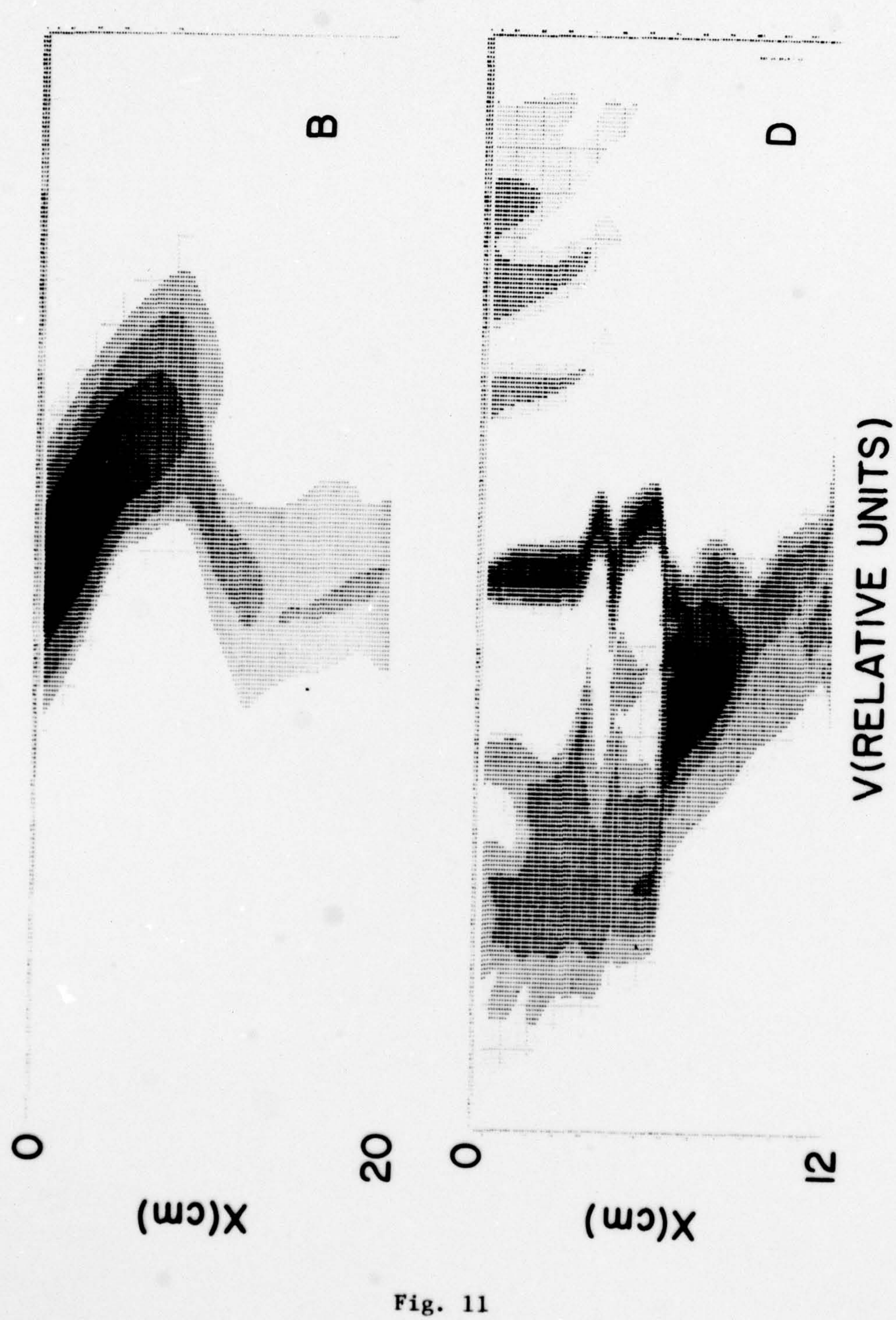


Fig. 11